DOCUMENT RESUME

ED 479 462 IR 022 138

AUTHOR	Gilbert, Juan E.
TITLE	Playing Teacher.
PUB DATE	2002-10-00
NOTE	7p.; In: E-Learn 2002 World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education. Proceedings (7th, Montreal, Quebec, Canada, October 15-19, 2002); see IR 022 093. Figures contain illegible type.
AVAILABLE FROM	Association for the Advancement of Computing in Education (AACE), P.O. Box 3728, Norfolk, VA 23514. Tel: 757-623-7588; e-mail: info@aace.org; Web site: http://www.aace.org/DL/.
PUB TYPE	Reports - Research (143) -- Speeches/Meeting Papers (150)
EDRS PRICE	EDRS Price MF01 Plus Postage. PC Not Available from EDRS
DESCRIPTORS	*Animation; *Computer Assisted Instruction; *Computer Simulation; Computer System Design; *Educational Technology; Instructional Materials; Programmed Instructional Materials; Teacher Role; Teaching Models; Video Games

ABSTRACT

The acceptance of animation technologies is increasing. Video games, such as Sony PlayStation (SONY, 2002), have become part of the culture for young people from kindergarten through undergraduate school. Animation technologies have been implemented into educational systems in the form of animated pedagogical agents (Johnson, 2000). The research described in this paper aims to take advantage of this widespread acceptance of animation technologies by introducing animated pedagogical agents as teachers in an adaptive instruction environment called Adam. Discussion includes the system's architecture, instruction creation, and instruction delivery. (Author/AEF)

Reproductions supplied by EDRS are the best that can be made
from the original document.

PERMISSION TO REPRODUCE AND DISSEMINATE THIS MATERIAL IN MICROFICHE ONLY HAS BEEN GRANTED BY

G.H. Marks

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

2B

Playing Teacher

Juan E. Gilbert[1]
Auburn University
Computer Science & Software Engineering
107 Dunstan Hall
Auburn, AL 36849 USA
gilbert@eng.auburn.edu

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

Minor changes have been made to improve reproduction quality.

Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Abstract: The acceptance of animation technologies into our culture is growing within the up coming generations. Video games such as Sony PlayStation (SONY 2002) have become part of the culture for young people from kindergarten through undergraduate school. Animation technologies have been implemented into educational systems in the form of animated pedagogical agents (Johnson 2000). The research described here aims to take advantage of this wide spread acceptance of animation technologies by introducing animated pedagogical agents as teachers in an adaptive instruction environment called Adam.

ED 479 462

Introduction

Gilbert (1999) discussed the implementation of an adaptive instruction system called Arthur. In this paper, Gilbert introduced the Many-to-One (M-1) instructional model. The M-1 instructional model refers to the relationship between educators and learners. In this model, there are many educators per learner. In order to fully understand this model, some terminology must be introduced. In this paper, courses are composed of many lessons. As an example, on a typical syllabus, there are many lessons listed that will be discussed through out the course. Lessons can be further divided into one or more concepts. Concepts are the smallest unit of instruction. These concepts are called instructional units through out this paper. An instructional unit has two subsections, instruction and assessment. In other words, each instructional unit has instruction, which is followed by an assessment for that instruction. Figure 1 gives an illustration of the M-1 instructional model.

In figure 1, there are three educators, Lopez, Brown and Smith, that teach the same course. This course has five instructional units. The rows represent a single course that is taught by the educator on the left. Therefore, figure 1 contains three implementations of the same course taught by three different educators. Each educator uses his or her own pedagogical methods to deliver instruction. The columns in figure 1 represent instructional units. Each instructional unit consists of instruction and an assessment created by the instructor at the top of the column. This implementation is how the M-1 instructional model begins. Another way to conceptualize the M-1 instructional model is to imagine a single learner in a classroom full of educators that teach the same subject, but each educators teaches the subject using a different pedagogical approach. In the physical implementation, the M-1 instructional model is not feasible. Using technology, such as the internet, it is possible to implement a representation of the M-1 instructional model. This implementation will be called Adam.

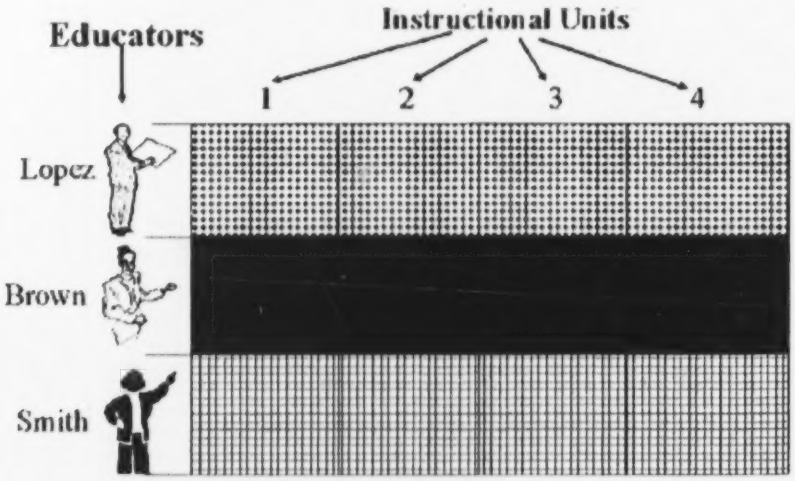

Figure 1: Many-to-One Instructional Model

IR022138

BEST COPY AVAILABLE

Architecture

The physical architecture of the Adam environment begins with a web server and an instruction repository. This web server is the agent server. The agent server will host all of the instructional units that were described in the previous section. The instructional units are stored in an instruction repository. Each instructional unit is maintained as a web deliverable, instructional screenplay. The instructional screenplays will be discussed in further detail in the instruction creation section below. Instructional meta data such as the instructional unit's name, description, author's contact information, etc. will be stored in the repository for each instructional unit. This data comes alive when it is accessed by one of the delivery systems.

Delivery Systems

Delivery systems can exist on different web servers or they can reside on the agent server. Delivery systems select instructional units from the instruction repository. This process is called instruction method selection, which is discussed in further detail in the instruction delivery section. Recall that each instructional unit is composed of instruction and assessment. The instruction has been created by an educator using his or her own pedagogical style(s). When a learner uses Adam, the learner will be assigned an instructional unit from the instruction repository. The process of selecting an instructional unit, instruction method selection, occurs inside of the delivery systems. Figure 2 gives an illustration of how delivery systems interact with the agent server and the learners. Notice that there can be more than one delivery system.

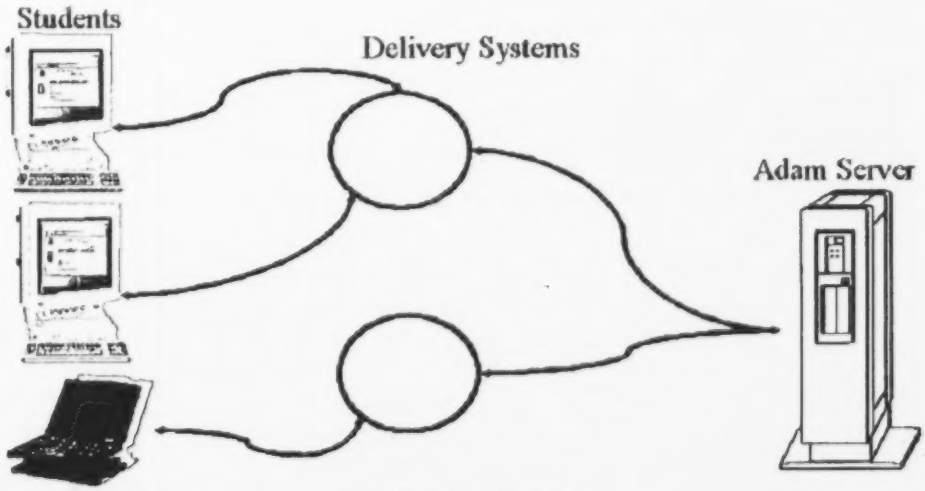

Figure 2: Adam Architecture

This is due to the fact that instruction method selection may vary. For example, in figure 1 the instructional units are numbered 1 through 4. The sequence for selecting these units may be 1, 2, 3 and 4 for a specific delivery system. For another delivery system, the sequence may be 2, 1, 3 and 4. This is an example of curriculum sequencing, which is the order in which instructional units are selected. Curriculum sequencing is just one variable that distinguishes delivery systems. The other variables will be discussed in the instruction delivery section. The server side of Adam has been discussed in this section and the user interface or client side follows.

User Interface

The user interface can be divided into two sections, navigation and content. Figure 3 has an illustration of the user interface layout. On the right hand side of the user interface there is a navigation frame. The navigation frame consists of a control panel and an animated pedagogical agent (Johnson 2000). The control panel allows the learner to start and stop a session, submit a question, logout, change the teacher's persona and to review previously seen instructional units. When the learner starts a session, the agent will begin to deliver instruction by speaking and manipulating web pages and/or objects in the content frame. The content frame is where all manipulatives used during instruction will appear. The agents used in this environment are implemented in the form of a web browser plug-in. This plug-in was developed by Haptek (2002). The

BEST COPY AVAILABLE

Haptek plug-in is downloadable and works on windows based machines. The plug-in performs text to speech in English and Spanish. It also provides voice manipulation and face animation. The face animation controls allow the developer to manipulate the agent's emotions in real time. The agent plays the role of educator in an elaborate educational screenplay.

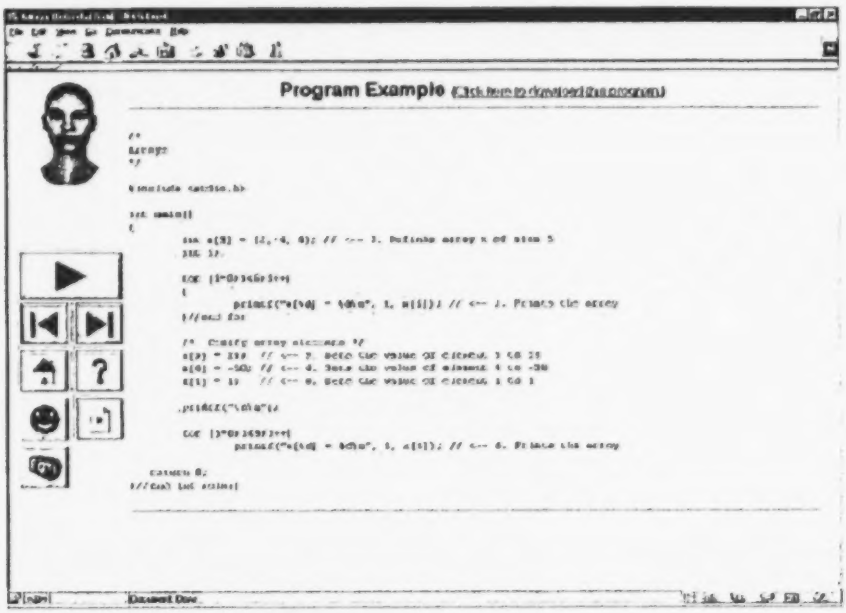

Figure 3: User Interface Layout

Instruction Creation

Before learners can use Adam, instruction must be created and implemented into Adam's architecture. The process of instruction creation is very tedious in its current state. The output of the instruction creation process is an educational screenplay that stars animated pedagogical agents. The process begins with real educators. In the current implementation, educators are video taped in a tutoring environment. The materials and all other manipulatives used during the tutoring session are collected when the session ends. Each session is focused on a specific concept. Recall, that concepts are later transformed into instructional units. The educators are also asked to create an assessment in the form of a quiz for each session. The goal of the assessment is to determine if the learner understood the concept(s). This process will be executed for a number of concepts with different educators. Once all of the concepts and assessments have been collected, the conceptual level of a Many-to-One instructional model can be implemented. In figure 1, there are three educators and four instructional units. Those instructional units would have been captured on video tape and the assessments created by each educator for those concepts will be collected as well. After all the concepts have been video taped from each educator and all the assessments have been collected, the instructional units will be created in the form of screenplays.

Creating Screenplays

The video tape of the tutoring sessions will be used to extract the educator's words as a digital audio file, .wav format. The .wav files of the educator's instruction will be transformed into compressed digital audio files that contain face animation information using Haptek's People Putty product. The output of this process is a .ogg file. In addition, each manipulative that the educators used will be transformed into a web deliverable format, i.e. html, video, Java, Flash. For example, everything that an educator writes or manipulates during the tutoring session will be transformed into a web compliant format. Next, the .ogg files and all the web deliverable materials will be placed along a conceptual timeline. This timeline is a reenactment of the tutoring session minus the physical presence of the educator and the learner. For example, an educator begins the tutoring session with a verbal introduction of a concept. This introduction may last 20 seconds. At that time, the educators writes a couple of sentences on a piece of paper or a chalk board. These actions would be replicated along

BEST COPY AVAILABLE

the conceptual timeline. Once the audio and materials have been conceptually synchronized, the educational screenplay will be stored in the instruction repository.

Storing Screenplays

The educational screenplays are stored in the instruction repository as instructional units. Recall, that instructional units are composed of instruction and assessment. From each tutoring session with a real educator, instruction is captured in the form of digital audio and web deliverable materials along a conceptual timeline. The assessments are also collected and converted into a web deliverable quiz. The instruction repository will house the digital audio/.ogg files, web deliverable materials and the conceptual ordering of the two together. The location or web address of the web deliverable materials is stored in the database on the agent server. This allows each web deliverable manipulative to be accessed via its web address, or url. This instruction creation process is very time consuming and labor intensive. In an effort to automate this process, an authoring tool is being developed.

Instruction Delivery

After the instructional units have been created and stored as screenplays with assessments, learners will be given access to them. When a learner uses Adam, the first task presented to the learner will be the educator selection screen. Figure 4 has a sample educator selection screen.

In figure 4, there are six different animated agents that can play the role of educator. Each agent has a different persona. Each agent has a unique persona that separates it from the others. Personas can differ in appearance and facial expression. Educator selection is similar to persona selection seen in popular video games such as Sony PlayStation (SONY 2002). In the video games, each player selects a team or person to represent him/her in the game. In Adam, the learner selects a teacher to provide instruction. The instruction behind the agent is stored on the agent server in the form of instructional units. Once the learner selects an agent educator, an instruction unit will be selected from the instruction repository.

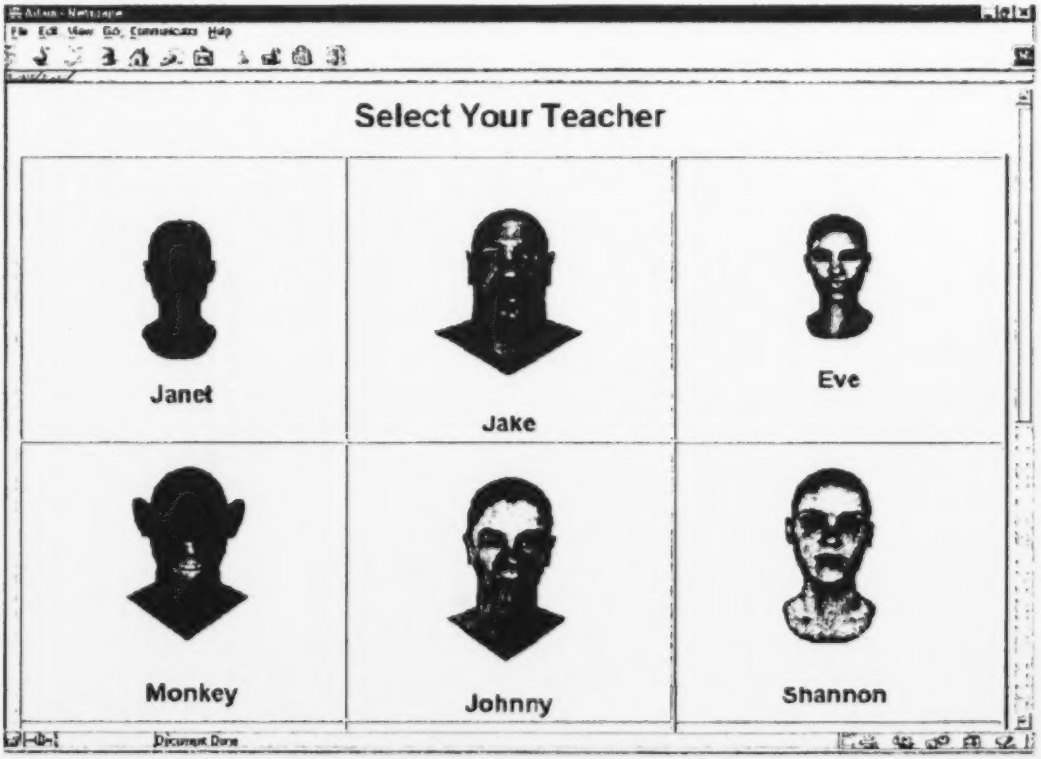

Figure 4: Educator Selection

BEST COPY AVAILABLE

Instruction Method Selection

The process of selecting an instructional unit from the instruction repository is called instruction method selection. Instruction method selection occurs at two distinct points within Adam, entry and after assessment. The first is the point of entry. When the learner selects an agent teacher, instruction method selection must occur. For example, learner Kristian logs into Adam. When Kristian enters Adam, a delivery system will be assigned to his course. Next, Kristian is presented with the educator selection screen in figure 4. He selects the first agent, Jake, along the bottom row to be his teacher. The system will refer to the instruction repository and select instructional unit number 1 from figure 1. Because Kristian is new to Adam, the first instructional unit will be selected at random. Notice that in figure 1, there are three different instructional unit number 1s. Assume Ms. Boykin's, the teacher in the middle, instructional unit 1 was selected. After the instructional unit has been selected, the interface will change and Kristian will be presented with the interface seen in figure 3 with Jake as the teacher. When Kristian presses the Play button on the control panel, Jake will begin to speak the transcribed text recorded from Ms. Boykin's tutoring session. The content in the right frame, figure 3, will change and display the web deliverable versions of Ms. Boykin's materials. This is the educational screenplay of instructional unit number 1 modeled after Ms. Boykin's tutoring session taught by Jake. Upon completion of instructional unit number 1, Kristian will be given the assessment created by Ms. Boykin for instructional unit number 1. After Kristian completes the assessment quiz, instruction method selection will occur again.

The completion of each instructional unit is marked by an assessment quiz. If the learner performs well on the assessment quiz, then the learner is presented with the next logical instructional unit using the same educator. For example, if Kristian performs well on the assessment quiz for instructional unit number 1, then he will be presented with instructional unit number 2, the next logical instructional unit, created by Ms. Boykin. In general, performing well is observed when the learner scores above a certain threshold. This threshold can be called the performance threshold. This may vary across delivery systems. For example, Gilbert [4] used a performance threshold of eighty percent for his studies. In other words, the learners had to score eighty percent or higher on their assessments before they could move onto the next logical instructional unit. Eighty percent was selected based upon mastery learning theory (Bloom 1976), where an eighty percent score determines mastery of a subject. Other delivery systems may set the performance threshold higher, i.e. ninety percent, or lower, i.e. seventy percent. In either case, the performance threshold must be met or exceeded before the learner can proceed onto the next logical instructional unit. If the learner scores below the performance threshold, then the learner will receive corrective instruction, which requires instruction method selection. Corrective instruction can be viewed as instruction that covers the same concept, but it is explained a different way. Therefore, the same subject is taught using a different instruction method. This process of providing corrective instruction when the learner falls below the performance threshold is called adaptive instruction.

Adaptive Instruction

Adaptive instruction refers to the process of adapting the instruction method in order to provide the learner with a personalized instructional experience. In some cases, adaptive instruction aims to accommodate learning style (Dunn 1978). In either case, adaptive instruction requires that an instruction method will be selected from a pool of varying instruction methods as seen in figure 1. Gilbert(2000) used a form of case-based reasoning to provide adaptive instruction in his studies. Each assessment quiz was defined as a case in those studies. All the cases consisted of similar attributes such as title, number of questions, number of questions answered correctly, etc. When a learner performed poorly on an assessment, the learner's assessment quiz would be transformed into a case. This case would be compared to previous learner cases. If a previous learner has performed in a similar manner as the current learner on the same assessment quiz, then the system would assign the current learner an instruction method based upon the previous learner's experience. This concept is based upon the premise that learners that perform similarly on the same assessments, may have similar learning styles. Therefore, establishing assessment quizzes as cases affords the ability to compare learners' assessments. For example, assume Kristian scored a seventy percent on the assessment quiz for instructional unit number 1 where the performance threshold was set to eighty percent. This quiz consisted of ten question and Kristian missed questions 5, 9 and 10. Also, assume a previous learner, Deborah, scored seventy percent on the same quiz, but she missed questions 5, 8 and 9. In the database, there is a case for Deborah and Kristian. Deborah's case contains information that shows she later passed instructional unit number 1 using Mr. Gabriel's instructional unit. Therefore, the system would assign Kristian Mr. Gabriel's instructional unit number 1 because Kristian and Deborah's assessment cases are closely matched and Mr. Gabriel's instructional unit number 1 worked well for Deborah. Adaptive instruction can occur throughout the learner's travel through Adam. An end result once the learner completes a course will be a learner map. The learner map represents all of the various instructional units selected by Adam. It is possible to compare learner maps at the end of the course in order to identify similarities between learners.

Conclusions

With the acceptance and integration of animation technologies into society, Adam has the opportunity to make a significant impact in providing universal access to alternative forms of instruction. As technology integrates further into the common household, tools such as Adam will give access to instruction from educators that would normally be out of reach. Adam has the ability to provide culturally responsive instruction to learners where their culture is not represented. For example, rural communities will be given access to instruction created by educators from across the globe. The demographic limitations of instruction are broken down by Adam.

Adam is part of an extensive research study that is supported by the National Science Foundation. Adam's instruction repository is currently being populated with instruction. The first prototype of Adam is being tested using introductory level computer science programming. On the instruction creation side, an authoring tool will be developed that will allow educators to create educational screenplays online. This tool is being called Eve. The tool will feature an animated pedagogical agent that plays the role of instructional designer. The agent will assists educators during the creation of their instructional units as screenplays. Educators will record their voice for instruction, which will be spoken during the screenplay. The educators will also have the ability to type in text that will be displayed on the screen and upload other web deliverable objects, i.e.: images. All of their materials will be stored on the agent server. The agent instructional designer will guide the educators through this process. It is estimated that Eve will be implemented within the next two years.

References

1. Bloom, B.S., *Human characteristics and school learning*. New York: McGraw Hill, 1976.
2. Dunn, K., and Dunn, R., *Teaching students through their individual learning styles: A practical approach*. Reston, VA: Reston Publishing, 1978.
3. Gilbert, J.E. and Han, C.Y. Adapting Instruction in Search of "A Significant Difference". *Journal of Network and Computing Applications, 22, 3.* (1999), 149-160.
4. Gilbert, J.E. and Han, C.Y. Researching Adaptive Instruction. Brusilovsky, P., Stock, O. and Strapparava, C. (eds.), Lecture Notes in Computer Science , 1892, *Adaptive Hypermedia and Adaptive Web-Based Systems*. pp. 409-414, 2000.
5. Haptek. Available at http://www.haptek.com/.
6. Johnson, W.L., Rickel, J.W., and Lester, J.C. Animated Pedagogical Agents: Face-to-Face Interaction in Interactive Learning Environments. *International Journal of Artificial Intelligence in Education 11* (2000), 47-78.
7. SONY PlayStation North America. Available at http://www.scea.com/.

[1] The research of this author is supported under the National Science Foundation grant #EIA-0085952.

U.S. Department of Education
Office of Educational Research and Improvement (OERI)
National Library of Education (NLE)
Educational Resources Information Center (ERIC)

NOTICE

Reproduction Basis

[X] This document is covered by a signed "Reproduction Release (Blanket)" form (on file within the ERIC system), encompassing all or classes of documents from its source organization and, therefore, does not require a "Specific Document" Release form.

[ ] This document is Federally-funded, or carries its own permission to reproduce, or is otherwise in the public domain and, therefore, may be reproduced by ERIC without a signed Reproduction Release form (either "Specific Document" or "Blanket").